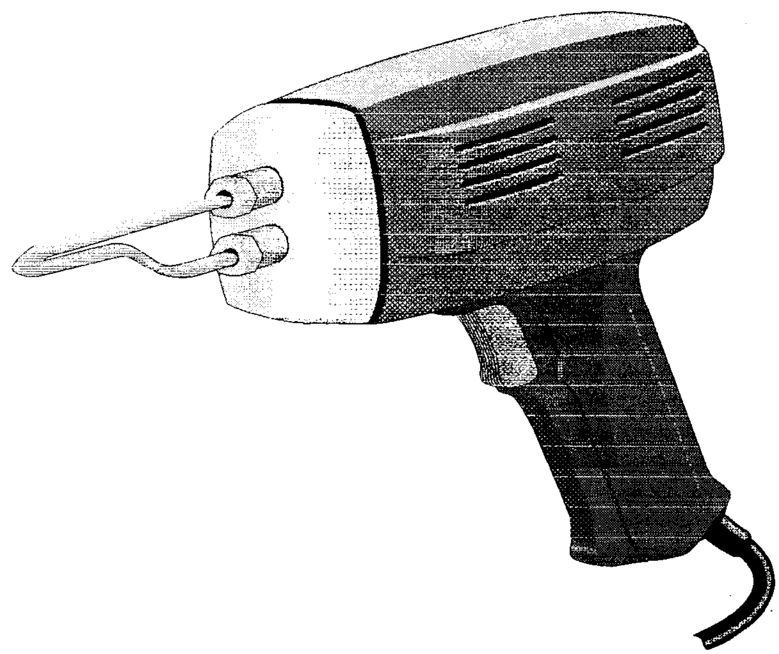
Issue: Winter 1996
$2.50

HBO ATARI USERS' GROUP

GAZETTE

The Hamilton, Burlington, Oakville ATARI Users' Group Newsletter for 8-Bit and ST Atari Systems



ATARI

Putting the Heat To Your Atari!

Executive for 1995

President
William M. Moeller 905-544-3387

ST Vice President
Peter Lorimer 905-634-7654

8-Bit Vice President
Larry Popa 905-388-4784

Treasurer/Secretary
Randy Charlebois 905-545-1179

ST Librarian
Roger Perrie 905-578-9179

8-Bit Librarian
Randy Charlebois 905-545-1179

Correspondent
Glen Atwell 905-549-9078

Editor
William M. Moeller 905-544-3387

Members at Large
Steve Kinsella 905-575-9164

The H.B.O. Gazette is a publication of the Hamilton, Burlington, Oakville, Atari Users' Group P.O. Box 35543 StrathBarton Postal Outlet, Hamilton, Ontario L8H 7S6

Membership

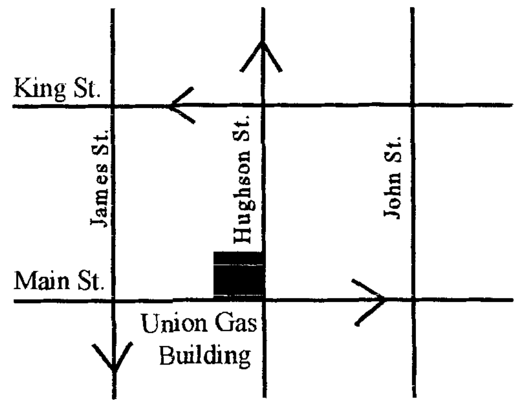
Membership Dues are $20 per person per year or $3 per meeting. The first meeting attended is free! The executive reserve the right to waive this policy for specified meetings. Membership includes a Gazette subscription and enhanced access to the club BBS, H.B.O. Online at (905)573-3704 and preferred pricing of the club public domain disks.

Meetings

Meetings are held in the Blue Flame room in the Union Gas Building at 20 Hughson Street South in Hamilton at 7:30 on the Second Tuesday of each month from September to June. There are no summer meeting nights.

Our Location

N



Contact People

General Information
Roger Perrie 905-578-9179

Atari 8-bit Programming
Larry Popa 905-388-4784

Atari 8-bit Hard disks
William Moeller 905-544-3387

Word Processing
Vacant 905-555-5555

Telecommunications
Glen Atwell 905-549-9078

Hardware Repair
Randy Charlebois 905-545-1179

Atari ST Programming
Vacant 905-555-5555

Atari ST Hard Disks
Roger Perrie 905-578-9179

Midi/Music
Vacant 905-555-5555

Desktop Publishing
Peter Lorimer 905-634-7654

Message from the President

Happy New Year! I hope everyone had an enjoyable Christmas 1995. This month we choose a new executive. I hope that we will see some new faces in the executive for 1996.

I would like to thank each and every executive member, and ordinary member, who helped run the club this past year. There is a lot of work to organizing interesting and informative meetings, distributing shareware, and of course, bringing you the Gazette you are holding in your hand.

We have come a long way this past year. Our club has improved. Our membership has grown.

However, there are still some areas that can be improved. We have had non effective Atari ST leadership this past year. We need someone to come forward to organize ST events aud presentations. We need some people to help out at General meetings, to take some of the burden off of executive members so they can enjoy the meetings also. If you are approached in the coming months to help with such things as writing a regular column for the Gazette, greeting new members, or helping with the club libraries, I hope you will help.

I would like to thank you for your vote of confidence last year in electing me president of the H.B.O.A.U.G . I have enjoyed working with the 1995 executive, and I am sure that the 1996 executive will be just as enthusiastic and hard working!

Sincerely

William M. Moeller
President H.B.O.A.U.G.

Next Meetings:
Tuesday February 13, 1996
Tuesday March 12, 1996

Table of Contents

Message from the President 2

H.B.O. Happenings 3

Hack Night For the Atari ST Series Computer 3

Atari Video Gaming Scene 4

H.B.O. Treasurer Report 6

Stabilizing Atari Computers with PBI Devices 6

Upgrading the memory of an Atari 600xl 8

The ST Librarian's Report 8

8-Bit Librarian Report 9

Do It Yourself Black Box Print Buffer 9

Memory Upgrades For Everybody 10

Transporting Atari Programs To The Atari 5200 . 11

H.B.O. Classified Ads 15

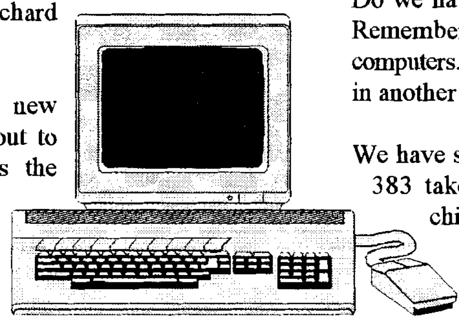
H.B.O. Happenings

By **William M. Moeller**

November 1995 saw another HBO swap meet. This informal meeting had a moderate amount of transactions, but was generally successful. Perhaps having a swap meet coupled with other events would better serve our members in the future.

December was another informal meeting. This traditional games night did not go as well as expected but was still a lot of fun. Atari ST computers were brought in for MIDI maze. Unfortunately, our ST librarian was not able to attend and no one had brought a copy of the program! However, two Atari Jaguar 64 video game systems were networked together for multi-player DOOM (tm). No one cared to keep track of high scores and claim a prize. All in all, it was still a really fun night for those who took part.

Richard Horachek has been bringing in the club book library the past few months. The club has an excellent selection of Atari specific books. If you would like to borrow or donate a book, see Richard during a meeting.



We have been having many new beginner Atari owners come out to our meetings. This highlights the need for our club to have a new user guide for both the Atari ST and 8-bit lines of computers. This project needs to be completed before the summer break. If you would like to volunteer, please see a 1996 executive member.

STATE OF THE CLUB BBS: Roger S Perrie Sysop

As you well know, if you own a modem that is, the HBO ATARI ONLINE BBS is more than just a little bit out of touch with reality. In the new year, I am hoping to change the software from Turbo BBS to BBS EXPRESS. I have an older copy of this BBS and the author has kindly released it as PUBLIC DOMAIN one complete version. There are lots of add-ons for the Public Domain ver. It will just take a little while to get it set up. Hopefully again, with a slightly larger Hard Drive so we can get a better upload/download area up and running.

Call H.B.O. Online at 573-3704

Hack Night For the Atari ST Series Computer

by Roger S Perrie

What a great surprise for a night at the computer club! HACK NIGHT! What kind of HACK are you interested in? The Library has several different ST hack files, from quieting down the ST WHINE, to installing a cooling fan.

Do you want to open up your serial ports to 38,400 baud? Do we have the software to support the modification? Remember, we use those outdated and obsolete computers. Sort of like the Pentium. It should be obsolete in another two or three hours!

We have several files on memory upgrades, HBO disk 383 takes in the latest ones I believe. You can add chips to your hearts content. I've used three different add-on memory boards for the ST and don't recommend any of them.

As a matter of fact, I'm up for offers on a 2 MEG board and two 2 or 4 MEG boards, minus the Ram chips. I've already soldered the chips into another circuit. The next upgrade I'm doing to an ATARI is straight to the Simms modules, and that upgrade is on the HBO disk.

As for Accelerators, we have two types:
Software Accelerators and Hardware Accelerators. The software type are along the lines of Warp_9 and NVD. They only give the illusion of speed by making the screen redraws work faster. Mind you, I wouldn't be without Warp_9, and it will work with the hardware boards as well. A good software accelerator is to be found in the $50.00 to $80.00 range, and in this area of the country are strictly a MAIL ORDER option on your part.

Mind once again, anything you want for any ATARI is a

MAIL ORDER thingy for anyone living in the "GOLDEN HORSESHOE". If you have the money for it go for a hardware upgrade. You are looking at the cost of the accelerator ($89.95 to 129.95 and up), plus the cost of a new set of TOS chips. You need them to make the new hardware work right! (About $79.95 USA) and of course, if you go this far you need a new AJAX chip to access a brand new Hi-Density Floppy Drive or two.

The SHAREWARE library has several text files on how to put in your own AJAX chip and installing Hi-Density Disk Drives. I have not yet found one that seems to be fully guaranteed not to eat my machine, or destroy the other devices plugged into it.

We do have on file in the Shareware Library file on how to do it such as!

Fixing a broken external power supply......

Fixing a non-working external disk drive....

Replacing Single Sided drives with Double Sided ones...

Adding a new AJAX chip for Hi-Density Drives....

Upgrading your serial port to 38,400 baud......

Plug into an older NTSC monitor, color or b/w...

How to share one Disk Drive between two ST's..

How to repair an ATARI SM124 Monitor......

Of course the list is endless:

Atari Video Gaming Scene

By Larry Popa

Happy New Year Gamers!

I hope all the HBO Jaguar owners received some really cool carts and CDS for Christmas. The release of the CD unit followed by the new price of $135 for the Jaguar unit along with the steady flow of new titles are all evidence that '96 should be a good year for the Jaguar! As you all probably already know the Networking interface (named the Jag-Link) is also out and was demonstrated in the last club meeting, and of course I also demonstrated how to whip Bill's butt at Doom <GRIN>.

The new Super Pro Controller is also out and has even more buttons than the standard Jag controller. It has three extra fire buttons, and two left and right buttons located on the top of the controller.

This month I'll be giving out a few more Jag cheat codes and will review Flip Out and White Men Can't Jump.

Hover Strike:

Alternate External Monitor Frame - Right+C+1+4+5 at mission selection screen
Extra Lives - 3+6+9+# at mission selection screen
Invincibility - Down+3+4+6+7 at mission selection screen
Level Skip - 2+4+6+7+8+9 at mission selection screen
Night Missions - Up+A+B+C+7 during game
Rotate the Globe - 4 or 6 at mission selection screen
Secret Bonus Missions (all entered during mission selection screen) -
Level 1 Bonus: Up+2+3+6
Level 2 Bonus: Down+2+6+7+8
Level 3 Bonus: Right+3+5+6
Level 4 Bonus: Up+2+5+8
Level 5 Bonus: Unknown

Val D'Isere Skiing and Snowboarding

SecretOptionScreen - 4+0+8+5+7+4+1+4

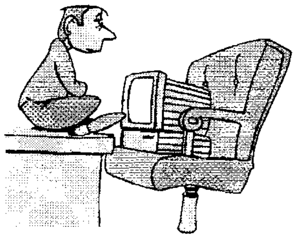
Wolfenstein 3D

Music Test - # at title screen
Level Select - 1+3+7+9 at main menu
God Mode - 4,6,6,8 during game
Full Inventory - 4,9,9,6 during game
Next Level - 4,7,8,6 during game
Previous Level - 4,6,9,6 during game
Refresh Info - 4,8,8,7 during game

Zool 2 (All done during Options screen)

Bulberry Hill - 7,7,7,4,9
Tooting Common - 8,8,5,6,3

Snaking Pass - 3,3,6,6,6
Mount Ices - 1,1,9,6,8
Mental Blockage - 9,1,2,6,6
Bonus Round - 3,1,8,6,7
Infinite Health - 1,1,2,7,5
99 Lives - 3,1,9,6,5



Flip Out! A Review

First Impressions

After my first attempt at playing Flip Out, the overwhelming initial impression I had was that this was a very odd game. When I played it a second time, I thought, geez, this really is an odd game. Once I tried it a third time I was convinced - Flip Out is an odd game. The idea is there is a set of nine tiles that have to be rearrange into the correct sequence. Sounds easy enough, right? Well, not really... The problem is you're given an extra tenth tile and if it ever touches the ground then it shatters and you have to start all over. As a result of this you have to keep this set of tiles up in the air and flip them around in the air to make them fall into the correct places. To really complicate things there are scads of really cute but annoying squishy blob-like creatures that play on the tiles while you're working with them. Some critters stand on the tiles and fly around on them, while others eat the tiles, and some are even cow-boys and buck around on the tiles saying, "Yeeeeeee haaaaaww!" Fortunately all of these cute little beasties make a satisfying squish sound when you kill them by smashing a tile down on their heads.... Muhahaha...

Graphics: ****

No complaints here. This is one good looking game. It's all done in 3D and the little critters are animated quite nicely.

Sound: **

The beasties make neat sounds, but it gets repetitive. The music is mediocre at best.

Game play: *

The Game play is terrible on this one. The manual gives very little information on the actual Game play and you're left on your own to figure out how to keep the tiles up in the air. The Game play is more repetitive than the sounds that the creatures make and does little to keep your attention. On the higher levels the beasties are so over-populated that the game becomes nearly impossible and you get so mad at the critters that you want to throw the controller through the TV.

Overall: 7/15

This game just doesn't do it for me. The graphics are amusing but once the effects of the graphics wear off there is nothing left to redeem this game. I could not fathom spending $72 for this game at Babbages and can only consider the purchase of this game to be a waste of money. I think I'll wait until this one hits the discount bin.

White Men Can't Jump:

First Impressions:

As the title implies this is a game based on the movie White Men Can't Jump. You borrow some cash from a local loan shark and invest it in conning other people out of money on the courts. Eventually when you have raised enough funds you go to the Slam City tournament for the big reward of $5000 if you win. Just like in the movie you are able to play better and do cooler moves when you are angrier. It's a great game with stunning 3D graphics and incredible sound. It comes packaged with a four player adapter called the Team Tap that allows you to have up to 4 players simultaneously for this game. The technique used to program this game entails constantly loading the information of all the various characters (for their animation) and decompressing it all in real-time as the game plays. This process can only be done on the Jag and as such a port of this game could not be done on the Play station or the Saturn even if they tried (take that Sony!)

Graphics: *****

The 3D manipulation of the characters, court, and background is mind-bogglingly unbelievable. The camera is constantly spinning and zooming in or out to always maintain the best angle for the shot. The animation of the characters is both smooth and plentiful in variety. There are over 25 different styles of slam dunks available. This game is a delight to watch in play.

Sound: ****

Good music for the most part and really great speech for the characters. Since this is street basketball the

characters shoot at each other and throw about various insults.

Game play: ****

This game is also quite fun to play. Being able to punch out the other characters is a nice touch. Having 25 different dunks available and many different characters adds a nice element of variety to the game.

Overall: 13/15

This game is well worth the $67 that Babbages wants for it. It comes with the Team Tap for free which sells separately for $32 at the Electronics Boutique. It's fun to play, and is still fun - even after you have beaten it. I highly recommend this one.

And that's all for this month. Enjoy the new year, and remember, "Read it, play it!"

H.B.O. Treasurer Report

By Randy Charlebois Secretary/Treasurer

H.B.O. ATARI USERS GROUP
FINANCIAL REPORT
December 1995

For the month ending December 1995 our fiscal balance was $193.54 with no outstanding debts. Our expenses for each meeting include pop , donuts and the newsletter, which come to approximately $40. Our revenue comes from memberships, raffles, 50/50 draws, and disk sales.

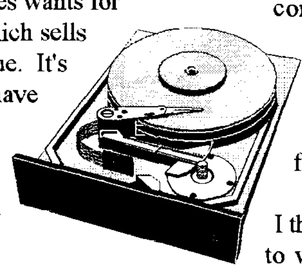
Help support the club by buying disks from the library and draw tickets. A volunter to run the draws and raffles would be greatly appreciated. If you have not renewed or joined the H.B.O. yet, why not? You will get your money's worth and more.

Our funds are adequate at the present to continue our club. More memberships and involvement by our members will allow us to support new and old users. Donations of Raffle prizes would be appreciated.

Stabilizing Atari XL/XE Computers with PBI Devices

By: Robert Puff 12/11/95

After having dealt with it seems thousands of Atari 8-bit computers that have come into my shop for repair, one gets to recognize some common ills, with the appropriate solutions. Recently, I have seen a number of computers that, by themselves, work fine; yet once they are connected to a Black Box, or large memory upgrade installed, suddenly they become flaky. Random crashes, bad bytes on the screen, and characters that have randomly changing pixels in them are some of the symptoms of this flakiness.



I think I've finally uncovered a solution that seems to work on just about every case. This solution is presented here, in hopes that this can help fix some 'field' problems, and give rock-stable performance.

Note that none of the modifications below will interfere with other peripherals, or cause any incompatibility.

Even if you are not using a PBI device, they can help.

This text file assumes the user is skilled with a soldering iron, and know how to identify and number the pins on an integrated circuit chip. If you are unfamiliar, let someone else do these mods for you!

Computer Software Services will perform the mods listed below for $25 plus shipping, if you so desire. Call (716) 429-5639 for information.

Although the 8-bits run at a very slow speed compared to today's machines, the bus timing is very critical. It seems that this flakiness is directly attributed to very small timing problems. While I can't say that I know 100% of what is going on, I'll try to explain what and where the problem is.

MODIFICATION #1

The 02 (phase two) clock signal that comes out of the computer through the cartridge (and PBI) port(s) is buffered from the signal coming directly from the CPU. This buffering adds a small but measurable delay. By the time the signal gets out of the computer and on to your

PBI devices, there is more inductance and parasitic capacitance to further delay this signal. When the delay gets too much out of hand, WRITES to the PBI device get corrupted, because the data bus is no longer valid on the trailing edge of 02. I used to swap out the 6502, which would generally fix the problem.

However, I found a solution that would work with *ALL* processors: add in the phase 0 clock input (that goes into the 6502's clock circuit) into the 02 buffer gate. the phase 0 signal is the same as the phase 2, only backwards in time slightly. It so happens that the 02 buffer gate is an AND gate, and has an unused input. Tying this unused input to the phase 0 signal ends up bringing the high-to-low transition back in time, giving us a little more grace for the extra delays that will happen in the outside world.

ON 600XL/800XL COMPUTERS:

Solder a wire from pin 4 of the 74LS08 to pin 13 of this same chip.

ON 65XE/130XE COMPUTERS:

Solder a wire from pin 2 of the 74LS08 to pin 4 of this same chip.

MODIFICATION #2

This modification deals with a timing problem with the OS ROM. It seems that especially with multiple EPROM OSes, the output buffers of the ROM chip stay on even into the start of the next cycle. This causes RAM corruption, easily seen by bad bytes randomly appearing on the screen. This fix isn't as simple as the first one. We need to connect our newly-fixed buffered 02 signal (from Mod #1) into the (inverted) output enable line of the OS ROM.

While one might think you could simply gate the chip select line with 02, better results seem to be had when you drive the ROM's chip select with the normal signal (coming from the PAL), and send the inverted buffered 02 signal to the output enable, which responds faster than the chip select pin.

ON 600XL/800XL COMPUTERS:

Solder a wire from pin 6 of the 74LS08 to pin 5 of the 74LS14 chip. (We're going to use the one unused gate in the '14 inverter chip.) The easiest way to finish this is to either desolder or cut pin 22 of the OS ROM, and bend the little stub up, so it is not making contact with anything. Solder a wire from pin 6 of the 74LS14 to this pin 22 of the ROM. If you have an Ultra speed + OS or some other sort of OS package and you can't lift this pin, you'll need to:

1. Cut the trace on the bottom side of the PCB tying pins 20 and 22 together.

2. Cut the trace that runs from pin 16 of the PAL (CO61618) to pin 22 of the OS ROM.

3. Run a wire from pin 16 of the PAL to pin 20 of the OS ROM.

4. Now run the wire from pin 6 of the 74LS14 to pin 22 of the OS ROM.

ON 65XE/130XE COMPUTERS:

You will need to add a 74LS14 chip.

Follow these steps:

1. Bend up all the pins of your new 74LS14 except 3, 7, and 14.

2. Stack this chip over the 74LS08 IC (oriented the same way), and solder pins 3, 7, and 14 of the two chips.

3. As in the XL instructions, there are now two options: lift pin 22 of the OS ROM, or cut traces. If you can just lift the pin on the ROM, then solder a wire from this lifted pin to pin 4 of your 74LS14. You're done!

4. Ok, so you want to do it the hard way! Actually, it's not that bad. Look at the bottom of the PCB. You'll see a trace that comes from the PAL, and goes to both pins 20 and 22. Carefully cut the small trace that goes to pin 22, leaving intact the one going to pin 20.

5. Now solder a wire from pin 4 of your 74LS14 to pin 22 of the ROM.

MODIFICATION #3

This is actually a Black Box modification. Again, due to varying phase two clock signals, a timing circuit on the

Black Box MAY need to be modified. The D1FF latch uses a R/C delay to insure the latching occurs while the bus is valid. A late 02 signal can skew this delay, causing the latch to grab random values at times.

R/C delay to insure the latching occurs while the bus is valid. A late 02 signal can skew this delay, causing the latch to grab random values at times.

The fix is simple. First, locate the resistor and capacitor combo that is just below the BB's SCSI port. Look at the color bands of the resistor. It should be brown-black-brown. Now look on the bottom side of the BB, and see if there is another resistor soldered in the same place. If so, then no modification is needed. If you see no resistor on the bottom *IN THAT LOCATION*, then solder a 220 ohm (red-red-brown) resistor across the two resistor pads.

Upgrading the memory of an Atari 600xl Computer

I just acquired a 600XL and want to boost the ram (it's currently at 16K). How do I go about doing this?

Going to 64k is easy. Remove the 4416 Ram chips in U12 and U11 and replace them with 4164 Ram chips. Next find U18, a 74LS32 and pull it out. Bend out pins 8,9,and 10 and reinsert the IC. Turn the board over and solder in a jumper wire from pin 8 to pin 10 of the same IC.

Next pull both the 74LS158's at U5 and U6, bend up pin 3 on one and pin 10 on the other. The chip with pin 3 bent up goes into U5, the one with pin 10 bent up goes into U6. Finally solder a jumper wire from the bent up pin 3 to U2 pin 4, the bent up pin 10 to U2 pin 5. That's it. If soldering onto the chip at U2 makes you nervous, you can follow the traces from U2 to a pair of solder pads you can use instead. The pads are to the left of U15 right next to pin 16, aprox 1/16 of an inch apart, one on top of the other. Bent up pin 10 goes to the top pad, bent up pin 10 to the bottom pad.

Going beyond 64K is another matter I will defer to more knowledgeable folk.

Good Luck!

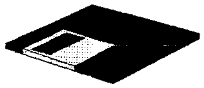
The ST Librarian's Report

By Roger Perrie
Club ST Librarian R.Perrie@Genie.Com

Hello to all my fellow club members here at the HBO, and of course to our "PREZ" Big BILL, who kept on my tail until I got this text file uploaded to him! The ST library is still evolving, from a 300 disk mess, to a 400 disk mess. Actually, a lot of the early disks had no documentation, all files were dumped onto the root directory, and an awful lot of them actually were pirated programs, (commercial software). Over the years, these disks also suffered from degeneration of the magnetic coating which makes the disk usable to the computer. This causes some loss of some of the programming on the disk itself. I have started to go through the disks, pulling all the files for each program and installing them into separate folders. These folders will then be "zipped" (archived) and put back onto fresh disks. I am in the process of separating these programs to different so called "THEME" disks. By this I mean, any given disk will contain similar programming. Eg, all games, or all modem programs, Maybe one or more Word Processors. By keeping them in an ARCHIVED format I can get more of them on any one disk, thereby giving you a better deal than if there were just one or two programs on the disk like you find in "COMMERCIAL" Shareware disks.

Hopefully, before the summer break, I will be able to release the Library's contents on disk for each member to check out while at home. I hope that by doing this it will allow everyone to make a more in-depth choice from the Library itself. I am also in the process of extracting as many files as I can from the CD- ROM set from Susy B's Software. I managed to borrow this two disk set and find it truly amazing. However, it does take forever to download the files from the CD-ROM, to my IBM, to 720k disks and then put them onto the ATARI to check out, re-arc, write a short bit about the program and then put it onto ATARI format 720k disks. Definitely a leading cause of the many, many white hairs to be found on my head!

As librarian, I don't have enough time to properly evaluate many of these programs and would like to see, either the ST Vice President, or one of the members-at-large take over some of the evaluation of these programs and to perhaps demonstrate one or two of these at meetings.



8-BIT LIBRARIAN REPORT

By Randy Charlebois

This month I'll cover the Utilities section of the library which covers Archivers, Copiers, Menus, Sector Editors and Upgrades.

For those with other computers (IBM) there are three disks available. XFORMER 2.0 and 2.5 cover emulating the 400/800, 800XL on your *OTHER computer*. There is also SIO2PC3 for using your IBM clone as a file server and Ramdisks for your 8-Bit. Check them out for your computers.

For those of you who have visions of upgrading your ATARI, the library now has the disks that come with these instructions.

We cover the 800XL up to 256k and the 130XE up to 1088k. For all the 8-bitters who are short on disk space, check out the Archivers. You can save 30% more files on the same number of disks. See our Copiers section with programs which will copy almost any disk you will ever want. Copy multiple files with one or two drives, or whole disks at one time by using extra memory.

Want to undelete a file, or personalize your disks? Check out UTL-300D D-UTILITIES for these and more disk utilities. Need to change a sector to repair or customize a disk? Look into the Sector Editor section.(UTL-500+). This section also includes a monitor and disassembler for those starting in machine language.

If you prefer running your programs from a menu look at UTL-101 for a collection of 15 handy disk menus.

There many other utilities in the library to do almost anything you could want. Check it out at your local H.B.O. 8-Bit PD library today! If there is a disk you want see the librarian at the meetings or give me a call before hand and I'll have the disks ready for you. There are a lot of good programs on these disks.

Do It Yourself Black Box Print Buffer

If you own a CSS Black Box without a printer buffer, Bob Puff has provided the following information to allow you to add the 64k printer buffer yourself.

This information is provided with the proviso that CSS not be held liable for any BB that gets damaged if you do not do the upgrade correctly.

If you have a BB ROM version 1.xx release, this modification with NOT work. Contact CSS for information on obtaining an upgraded ROM.

You will need two 32kx8 Static RAM chips (SRAMs): Sony CXK58256PM-12 or NEC D43256AC-12 or equivalent.

Jameco-Electronics sells both chips, although
NEC chip is cheaper.

Contact JAMECO at 1355 Shoreway Road, Belmont CA 94002

Do not buy the much faster SRAM chip used as "cache RAM" in MS-DOS machines such as an 61C256. These are the wrong size, among other things. You must have a 28-pin DIP that matches the 8Kx8 SRAM already installed in a stock BB (an Hitachi HM6264LP-15). You will also need a few lengths of insulated wire and the usual assortment of small hand tools to complete the job.

Stack the two SRAMS and solder all the pins together except pins 20 and 22. Raise pins 20 and 22 of the upper SRAM and tie them together. Remove the 8K SRAM from the Black Box and install this stack, insuring that the index notch in the end of the chips is pointing in the same direction as the original SRAM.

Now, locate the 74LS139 chip in the front right hand corner of the Black Box. Solder a wire between pin 5 of this chip and pins 20 and 22 of the SRAM stack. Lift pin 3 of this same LS139 chip and solder a wire between it and pin 9 of the 6821 chip on the Black Box (a 40 pin chip to the left of the SRAM stack).

That is all there is to it! Reconfigure your BB by going to the Port Statuses screen and changing the spooler stat line to read Internal RAM: your 64k printer buffer is now active.

MEMORY UPGRADES ARE FOR EVERYBODY

By Gary Ferris with John Nagy (C.H.A.O.S.)
Edited by Randy Charlebois December 1995

A great deal of attention has been focused on the large number of ATARI computer memory upgrades. This article is not intended to deal with the technical aspects of the various upgrades, but will try to point out the reasons why you should consider any particular memory upgrade.

The most common usage of extended memory is as a simulated disk drive, commonly called a ramdisk. The advantage of this usage is its incredible speed. Its main disadvantage is that the contents are lost when the machine is turned off unless first transferred to a real disk drive. Many additional uses (such as the fabulous BASIC XE from OSS, Inc.) are continuing to surface as extended memory becomes commonplace.

THE 800XL UPGRADES

If you are fortunate, all the chips will be in sockets. If so, then changing the required chips will be a snap. If not, then exercise great care in desoldering the chips and CAREFULLY inspect the double sided board for damaged paths. Repair any damage and install sockets. This will make it far easier to change any chips which may go bad in the future.

One of the first upgrades for the XL was invented by Claus Bucholz and published in BYTE MAGAZINE. His original design was not compatible with the 130XE, but Claus soon modified it to be XE compatible and placed the plans and source code for his RAMDISK drivers in the public domain. The necessary files are in the 8-Bit library under utilities.

THE RAMBO XL

It is an upgrade board by makers of SpartaDos and many fine pieces of hardware for the 8-bit ATARIs. I have personal experience with the Rambo XL it is a good quality product. It is an XE compatible upgrade and the installation is reasonably straightforward.

THE NEWELL UPGRADE

The upgrade board by Newell Industries is not fully compatible with any of the above upgrades but is more nearly compatible to the actual 130XE in that it retains the ANTIC enhanced mode. (The 130XE allows you to use the extra memory for the CPU or as video ram (ANTIC).) Since there is very little software which uses the ANTIC mode this is not an important consideration. For more info, contact Newell Industries (214)442-6612 or your local ATARI dealer. I'm not sure if this upgrade is still available. (Editors Note: I believe Fine Tooned Engineering has acquired all of Newell Industries Atari 8-bit products. See Fte interview in Christmas 1995 H.B.O. Gazette)

THE 600XL UPGRADE

Where the 800XL's use 8 64k by 1 bit chips, the 600XL uses 2 64k by 4 bit chips. This means that pile of old chips you've been saving is useless and also that the 256k upgrades for the 800XL will not work on the 600 XLs. I don't know if there are 256k by 4 bit chips available or not. If not, then 64k is the practical limit for a 600XL. It is an easy upgrade though requiring only the new memory chips and 3 jumpers. Most clubs have these plans available.

THE 320K 130XE

First, I'd like to thank Scott Peterson for sharing the upgrade instructions with ATARI users. It's a very clean modification since most of the circuitry for bank selection is already built in the XE. To do the 320K upgrade is very simple if you have any experience at all with soldering. I also ***HIGHLY*** recommend installing sockets in the spaces where you removed IC's (8 16 pin and 1 40 pin socket) They are not absolutely necessary but this is the ideal opportunity. If you ever have a chip go bad in the future it will be a lot easier to replace this way.

The entire 320k can be done in less than an hour (most of this time being in removing the old chips) Only one chip is required in addition to the 256k memory chips which are substituted for the second 64k bank.

THE 576K 130XE

The 576K upgrade is quite a bit more involved than the 320K. You have to undo some of the 320 upgrade, add 2 additional support chips, and piggyback 8 more 256K chips on the 8 previously installed. This involves a large number of jumper wires also. The additional time to do this mod was about 2 hours (so a total of about 3 hours to do the 576k at one sitting). You also lose your internal

BASIC ROM in the 576k mode. A switch is added which returns your internal BASIC but only gives you use of 192k. To use BASIC in the 576K mode requires a cartridge such as BASIC XE.

THE 1088K 130XE

This is the largest upgrade documented to date. In all honesty, the only reason I could see for doing it may be useful on a BBS. Another potential problem is that if you use items such as the XM301 and PR Connection which take their power from the computer, they may not work properly do to inadequate power supply capacity.

In my opinion, the best DOS available, especially for upgraded machines, is SpartaDos. The RD.COM file will automatically format the RAMDISK for whatever configuration you have (up to 576K) with the command RD Dn:. To reserve the extra 64K for Basic XE the command is RD Dn: /E (space between : / IS REQUIRED). You can also enable the Ramdisk without formatting by RD Dn: /N.

This would make it possible to copy a program and it's required data files to the ramdisk as D2:(or whatever), and then turn off the real disk drive and rename the ramdisk as D1:

Information on most of these upgrades as well as MANY different RAMDISK handlers can be obtained by checking with the librarian. Upgrade text files and programs are in the library.

Transporting Atari 8-bit Programs To The Atari 5200 SuperSystem

By Claus Buchholz
A.N.A.L.O.G. #15, January 1984

When Atari designed the 5200 "SuperSystem" as a successor to the aging 2600 VCS, they made use of the state-of-the-art hardware they put into their 400/800 home computers. As a result, the systems are quite similar. The differences are great enough, however, that transporting programs from one system to the other requires some effort.

The 5200 is a single-board machine with four controller jacks, a cartridge slot, an I/O expansion connector and a power/RF cable. It shares the same VLSI chips with the 400/800, except for the 6520 PIA (joystick/parallel I/O ports). The other chips (ANTIC, GTIA, POKEY and the 6502 CPU) are in the 5200, although some of their registers are in different memory locations than those in the 400/800.

Also, some of the registers serve slightly different purposes. Figure 1 is a block diagram of the 5200. Let's look at each section of the hardware in turn.

When a specific register is mentioned in the article, the name is taken from the Hardware Manual in the 400/800 Technical Reference Notes. With the exceptions listed in this article, the Hardware Manual applies also to the 5200.

6502 CPU

Although it is a standard 6502 from the programmer's view, Atari has reworked this chip to save four support chips. Those four chips mediate access of the system bus by the 6502B and ANTIC in the 400/800, but that function is built into the 6502C in the 5200. Atari also uses the 6502C in the new XL series of computers.

16K RAM

The 5200 contains 16K bytes of RAM addressed from $0000 to $3FFF, just as in an unexpanded 400. The memory circuit is nearly identical to that in the 400, except it uses 4516s, a 5V-only version of the 4116 16K-bit dynamic RAM.

The 5200 monitor program reserves locations $0000 through $0018 and $0200 through $021B for shadows and RAM vectors. And, of course, page $01 is reserved for the 6502 stack. The rest of the RAM is available to the cartridge program.

ANTIC

This is the same ANTIC chip used in the 400/800. Since ANTIC shares the address bus with the CPU and has no chip select input, its registers reside in the same place in the 5200's memory as in the 400/800's, page $D4.

The 5200 has no System Reset key, so bit 5 of NMIST is useless and the 5200's interrupt handler ignores it.

GTIA

The GTIA and its registers perform the same functions in the 5200 and 400/800, except as noted below. The registers, however, reside at page $C0 in the 5200's memory, not at $D0 as in the 400/800.

The trigger inputs, TRIG0 through TRIG3, are wired to the controller ports, one to a port. The bottom button on either side of the leftmost controller zeroes the TRIG0 register when pressed, and likewise for the other ports.

The bits in CONSOL, the 400/800's console switch port (START, OPTION, SELECT and speaker), are used as outputs in the 5200, Bit 3, the 400/800's speaker control can still be toggled in the 5200 to produce sounds through the TV speaker. Bit 2 controls the pots in the joystick controllers. It must be set high to enable the pots.

Bits 1 and 0 select which controller port is to be active at one time. 00 selects port #1 (the leftmost), 01 selects #2, 10 selects #3, and 11 selects #4. The trigger buttons and pots are independent of this selection; it applies only to the keypads and top side buttons on the controllers.

POKEY

POKEY's registers are all addressed at page $EB in the 5200 as opposed to $D2 in the 400/800. Its functions are unchanged, however, except for two.

The eight pot inputs used for paddles in the 400/800 are wired to the 5200's controller ports, two to a port. Each controller has an analog joystick, using one pot to sense horizontal position and a second pot for vertical position. The even pots (POT0-POT6) give the horizontal positions of range from 1 to 228; the maximum readings are to the right for the horizontal pot and at the bottom for the vertical pot. Figure 2 shows the pinout for the 5200 controllers.

Pin	Function
1	Keypad -- right column
2	Keypad -- middle column
3	Keypad -- left column
4	Start, Pause, and Reset common
5	Keypad -- third row and Reset
6	Keypad -- second row and Pause
7	Keypad -- top row and Start
8	Keypad -- bottom row
9	Pot common
10	Horizontal pot (POT0, 2, 4, 6)
11	Vertical pot (POT1, 3, 5, 7)
12	5 volts DC
13	Bottom side buttons (TRIG0, 1, 2, 3)
14	Top side buttons
15	0 volts -- ground

The keyboard scanning lines in the 400/800 are used in the 5200 to read the keypad keys to the one controller that is selected by bits 1 and 0 in CONSOL. Only four lines are used, though, so only bits 1 through 4 of KBCODE are valid.

Table 1 gives the keycode for each key on the controller. The top side buttons on the selected controller act like the 400/800 shift keys and also cause a BREAK-key interrupt, if that's enabled. Bit 0 of SKCTL, the debounce enable bit, need not be set in the 5200.

Key	KBCODE bits	Keypad code
none	0000	$FF
#	0001	$0B
0	0010	$00
*	0011	$0A
Reset	0100	$0E
9	0101	$09
8	0110	$08
7	0111	$07
Pause	1000	$0D
6	1001	$06
5	1010	$05
4	1011	$04
Start	1100	$0C
3	1101	$03
2	1110	$02
1	1111	$01

POKEY's serial I/O lines are used in the 5200, but they are wired to the I/O expansion connector, an edge connector hidden behind a small door in the rear of the 5200. This connector allows for more hardware registers addressed at page $E0, and for peripherals using the serial port. See Figure 3 for the pinout of this connector. Its existence demonstrates Atari's original plans to expand the 5200 system.

Top	Bottom	
+5V DC	1 36	+5V DC
Audio Out (2 port)	2 35	Not connected
Ground	3 34	Ground
R/W Early	4 33	Not connected
Enable E0-EF	5 32	D7
D6	6 31	D5
D4	7 30	D3
D2	8 29	D1
D0	9 28	Ground
IRQ	10 27	A0
Ground	11 26	A1
Serial Data In	12 25	A2
Serial In Clock	13 24	A3
Serial Out Clock	14 23	A4
Serial Data Out	15 22	A5
Audio In	16 21	A6
A14	17 20	A7
System Clock 01	18 19	A11

ROM

The 5200 has a 2K ROM on board which holds the character set and monitor program. The character set, which is an exact copy of the 400/800's set, resides at pages $F8 through $FB, and the monitor sits at $FC through $FF.

The cartridge ROM can be 32K bytes long and resides in memory from $4000 to $BFFF. Figure 4 shows the pinout of the cartridge slot. The two interlock connectors are wired together on a cartridge board. The 5200 uses this as a switch for the cartridge's power connections and as a Reset signal. Therefore, a cartridge may be safely removed or inserted while the 5200 is powered on.

D0	1 36	Interlock
D1	2 35	A11
D2	3 34	A12
D3	4 33	A10
D4	5 32	A13
D5	6 31	A9
D6	7 30	Audio In (2 port)
D7	8 29	A8
Enable 80-8F	9 28	Not Connected
Enable 40-7F	10 27	A7
Not Connected	11 26	+5V DC
Ground	12 25	Ground
Ground	13 24	Ground (Video In on 2 port)
Ground (System Clock 02 on 2 port)	14 23	Ground
A6	15 22	A4
A5	16 21	A3
A2	17 20	A1
Interlock	18 19	A0

The 5200 monitor program.

The 1K monitor program in ROM has three functions: (1) to initialize the system before running the cartridge program, (2) to service interrupts as they occur, and (3) to maintain shadows of some important hardware registers. Recall that the 400/800 Operating System is 10K bytes long and performs the above functions. It also provides peripheral handlers, predefined graphics modes, a screen editor, and floating point math routines. Those utilities do not exist in the 5200.

Table 2 shows the RAM locations used by the monitor for shadows and RAM vectors.

Table 2. 5200 Monitor RAM Locations.

Page Zero Locations

$00 Shadow for IRQEN
$01 Real time clock (high byte)
$02 Real time clock (low byte)
$03 Critical code flag (if non-zero, VBI routine is abbreviated)
$04 Attract mode timer/flag
$05 Shadow for DLISTL
$06 Shadow for DLISTH
$07 Shadow for DMACTL
$08-$10 Shadows for COLPMO through COLBK
$11-$18 Shadows for POT0 through POT7

Page Two Vectors

$200 Immediate IRQ vector
$202 Immediate VBI vector
$204 Deferred VBI vector
$206 DLI vector
$208 Keyboard IRQ vector
$20A Keypad routine continuation vector
$20C BREAK key IRQ vector
$20E BRK instruction IRQ vector
$210 Serial Input Data Ready IRQ vector
$212 Serial Output Data Needed IRQ vector
$214 Serial Output Finished IRQ vector
$216 POKEY Timer 1 IRQ vector

$218 POKEY Timer 2 IRQ vector
$21A POKEY Timer 4 IRQ vector

Upon Reset, the 6502 vectors through $FFFC to the initialization routine. This routine performs the following sequence.

1. Disable maskable interrupts, clear the 6502 decimal flag, and set the stack pointed to $01FF.

2. If the cartridge address $BFFD contains $FF, then jump immediately through the vector at $BFFE (diagnostic cartridge).

3. Zero all hardware registers and page $00, set CHBASE to point to the character set at $F8, and initialize the first six RAM vectors starting at $0200.

4. Set up the Atari logo rainbow display. The cartridge title (20 characters) and copyright year (2 characters) in ANTIC mode 7 display code are taken from cartridge addresses $BFE8 through $BFFD.

5. Enable VBI (Vertical Blank Interrupt) and DLI (Display List Interrupt), and enable key scan.

6. Wait four seconds, then jump through the vector at $BFFE to the cartridge program.

When the 6502 receives a non-maskable interrupt (NMI), it vectors through $FFFA to the NMI handler. The following steps take place:

1. Check NMIST and strobe NMIRES to reset the interrupt status.

2. If a DLI is pending, jump through the DLI vector (initialized to point to the rainbow effect routine).

3. If a VBI is pending, jump through the immediate VBI vector (initialized to point to the VBI routine).

4. Else, return from the interrupt (no System Reset).

A cartridge program can change these vectors to point to its own DLI and VBI routines, if it must. The default VBI routine takes the following action.

1. Push A, X, and Y onto stack, increment the real time clock, and update the attract mode timer.

2. If the critical code flag byte is non-zero, then pop Y, X, and A from the stack and return from the interrupt.

3. Update DLISTL, DLISTH, and DMACTL from their shadows.

4. Maintain the attract mode flag and update the GTIA color registers from their shadows.

5. Update the pot shadows from POT0 through POT7, and strobe POTGO to start another pot scan.

6. Jump through the deferred VBI vector (initialized to point to the end-of-interrupt routine, which pops Y, X, and A, and returns from the interrupt).

If maskable interrupts (IRQs) are enabled and one is received, the 6502 vectors through $FFFE to an instruction which jumps through the immediate IRQ vector. That vector is initialized to point to the IRQ routine, which performs the following tasks.

1. Push A and check IRQST.

2. For each of the eight bits in IRQST, check for a pending interrupt. If found, then clear the status bit, update IRQEN from its shadow, and jump through the appropriate IRQ vector.

3. If no interrupt found, then push X and check for a BRK instruction interrupt. If found, then jump through the BRK instruction IRQ vector.

4. Else, pop X and A and return from the interrupt.

The only IRQ vector that is initialized is the keyboard IRQ vector, which points to the keypad read routine. That routine does the following:

1. Push X and Y.

2. Read KBCODE and mask bits 1 through 4.

3. Convert to the keypad code given in Table 1, leaving that code in A.

4. Jump through the keypad routine continuation vector (initialized to point to the end-of-interrupt routine).

Comparing the 5200's monitor vectors to the 400/800's OS vectors, we see that Atari paid no attention to compatibility between the two. This further complicates

the task of converting a program from one system to the other.

Transportability

It would not be difficult, given the information in this article, to write a program in two versions, one for the 400/800 and another for the 5200. Nor would it be difficult, given the source code, to convert a finished program from the 5200 to the 400/800. The reverse is more difficult if the program takes advantage of special features in the 400/800 OS. Otherwise, the only task, aside from redefining some addresses, is to convert the keyboard/joystick input routines from one system to the other.

I acquired the information in this article by dissecting a 5200 and disassembling its ROM. The 400/800 schematics in the Hardware Manual were quite helpful. It is interesting to note the difference between the two machines and to guess Atari's motives for the design differences. But the similarities grossly outweigh the differences, so that a 5200 program can be developed and almost entirely debugged before testing on a 5200. With the addition of an EPROM burner, a 400/800 can be a powerful development system for 5200 programs. An adventuresome hacker can even bypass the EPROM by putting dual-port RAM on the cartridge board and downloading programs from the 400/800 development system into the 5200 for testing.

A.N.A.L.O.G. #16, February 1984

READER COMMENT

5200 Article Update. (ANALOG #15)

Newer releases of the 5200 incorporate some minor hardware changes. Controller ports 3 and 4 have been eliminated, making POT4 through POT7, TRIG2, TRIG3, and bit 1 of CONSOL useless. A few of the connector pins have been redefined. Pin 2 of the I/O expansion connector now carries POKEY's Audio Out signal. Three pins on the cartridge connector have changed to accommodate the new 2600 adapter. The system clock, 02, is output on pin 14, isolated through a diode. An alternate video input is taken from pin 24 and is also isolated through a diode. Pin 30 provides an alternate audio input.

There is space on the newer boards for circuitry for a PAL (European TV standard) version of the 5200. Also, on power-up, the monitor program checks for the PAL version by examining the GTIA register PAL after step 2 of the initialization routine. It also checks the cartridge program for PAL compatibility. The byte at $BFE7 should read $02 if compatible, or $00 if not. This is the only important change to the monitor program. There are some additional hardware changes, but none affects the machine's operation from the programmer's view.

Claus Buchholz
Greenwich, CT

H.B.O. Classified Ads

